Appendix 1

[01]   There are numerous configuration forms within the NMS 130 developed by many different developers, in many different groups. In the future even more diverse numbers of groups could develop features for the NMS 130 and be responsible for new configuration forms. It is essential for new development be undertaken in a consistent manner. Consistency presents a professional looking product to the analyst as well as reduce training times, allow operations within the NMS 130 to be undertaken in less time and limit the amount of customer support which vendors will be required to provide.

[02]   It is important that the order and layout of widgets within the window match the workflow of the analyst. Text fields and labels, widget selection, filtering tasks, etc., are to be positioned within the screen so that the analysts do not have to undertake excessive mouse movements or visual searches to accomplish tasks. The positioning and labeling of repeated items or item types within the different configuration forms are required to remain consistent between forms. Consistency between forms builds a confidence that will help analysts navigate through new configuration forms with fewer errors, at greater speeds and with less training.

[03]   Configuration forms are intended not to have menus and tool bars. All required commands for configuration forms should be accessed through the use of command buttons, either in a command button column or positioned next to specific areas within the form that is active. Command buttons provide the analyst with unhidden, accessible options to undertake actions within the configuration form.

[04]   Generic configuration windows should consists of the following standard items or areas:

[05]   The title bar contains the name of the configuration form and generally specifies the managed entity that is being configured.

1

[06]   This area contains information fields that, when filled in, will configure different parts of the network or different aspects of the network management and service provisioning solution. This section could contain standalone command buttons for undertaking specific tasks related to widgets within the view panel.

[07]   Generally, the configuration fields area should be presented as a tabbed pane that will allow view panel developers the flexibility of breaking up the fields that are required for a configuration into manageable amounts for the analyst. Also, it provides a possible simple way to add information to the form in the future.

[08]   Nested tabs within the configuration forms are not preferred. However, tabbed features within forms can be implemented without requiring the entire form to be tabbed.

[09]   The command button column should preferably have a vertical orientation and should be placed at the top right of the form.

[10]   Command buttons can be located outside of the Command Button Column. This allows buttons to be associated with the specific areas of the view panel that they influence.

[11]   A Message Bar, or text display, may be utilized for help or action notes specific to the configuration form which is being filled in. An example would be: if a value being entered into a field is out of range, it could be highlighted in this area.

[12]   A Progress Bar is preferably presented within the view panel to indicate that work is being done. Typically the progress bar is to be used for tasks that take 10 seconds or more, such as downloading information or filtering values.

[13]   Often features may be disabled (grayed out) or completely hidden to prevent analysts from accessing them (scope-of-command/span-of-control).

This happens for various reasons such as: not having appropriate access permissions to operate certain features, not having the input correct or enough information to undertake an action, not having checked off the preference which activates a feature etc.

[14] In general, if the analyst could under different circumstances use a feature, then it is appropriate to gray it out. The analyst will be aware of the feature's existence and understand that actions are required to get access thereto. Grayed-out features are also valuable because they allow the analyst who is exploring an unknown program to see that they exist and have functionality.

[15] If the analyst does not have access to features and would never have access or reason to use them, then it is appropriate not to show the feature at all. Grayed-out features which will never be used clutter the view panel, confuses the analyst and possibly increases the time taken to access the features which can be used (i.e., the analyst is required to scroll / tab though grayed-out menus, tabs and buttons etc.)

[16] The configuration view panels will typically use two forms of error warnings:

— When information is entered into or selected for a field and it is of the wrong type, the configuration form will not accept the entry and a message will typically be displayed at the bottom of the screen in a message bar.

- When incorrect values have been entered and the

[17] "Ok", "Apply" or any other confirmation button is clicked, an alert error dialog window may pop-up advising the analyst that problems have been encountered.

[18] Alert boxes convey messages and warnings to the analyst. They contain window titles, symbols, message headers, messages and command buttons.

[19] Error alert boxes display a message describing a major problem that has been encountered and requires the analyst's intervention or correction in order to continue. Error alert boxes contain an icon representing the severity of the problem, a textual description of the error, the error number in the title and typically one button. Error alert boxes may also contain controls to correct the error. If the error cannot be corrected from the dialog box, a Close button is preferably presented. A Help button is optional (when deemed appropriate).

[20] A button representing the corrective action may be presented if the analyst can correct the error, followed by an escape button such as Cancel, and finally Help (if deemed appropriate). Cancel would close the error alert box and cancel the operation that initiated the error.

[21] Error alert boxes like all alert boxes are created automatically by the framework 220.

[22] Other alert boxes can result from the use of configuration forms. Following is a brief list of the different types:

[23] Informative alert boxes are used only for conveying information and only contain a Close button. These display a body of text typically.

[24] Warning alert boxes prompt the user to confirm an operation. These contain a textual description and minimally two buttons. A first button typically is a Continue or Yes button that reflects the action about to be performed. A second button allows the user to back out of an operation and would typically be labeled Cancel or No, depending on the message context. If appropriate, a Help button can be added as a third button.

[25] A default button should not be set if the action about to be performed has potentially destructive consequences.

[26] Generic configuration forms may use two kinds of progress indications:

4

— A Progress Indicator Bar is typically located at the bottom of the configuration form. The progress bar typically appears while filter or look-up tasks are being performed.

— A Progress Dialog Box, would only appear after the configuration forms inputs have been applied though the use of an "OK" button (or equivalent). It is only to be used on those configuration forms that require a long period of time to implement previously entered information.

[27] A Progress Indicator is typically located on the right side of the status bar. A progress indicator is used in the status bar if the application performs frequent, non-interruptive operations, likely to last more than 10 seconds. A non-interruptive operation would be one that does not require the analyst to stop working until it is complete.

[28] If it is impossible to determine the progress of the activity towards completion (i.e., the completion point is unknown, as in a database query), then the use of a looping progress bar is preferred. A text pane can be used alongside the progress bar to give feedback on what the progress bar is indicating.

[29] If the rate of progress towards a known completion value is known, then the use of a non-looping Progress Bar is preferred. It is preferred to indicate the percentage complete within the progress bar itself. A text pane next to the Progress bar may be used to give additional information and feedback.

[30] A progress dialog may be used to modally display the progress of a process that the application is undertaking. It is to be used if the analyst cannot continue working until the process is done (i.e., they cannot multitask within the application). For example using a progress bar within the status bar of the primary view panel would be appropriate if the application has sent a query to a remote database, allowing the client machine to continue working. However, an action such opening a large file on the client machine would not allow for the

5

analyst to perform other tasks, so a progress dialog would be appropriate. Progress dialogs are to be used for expected delays of more than 10 seconds. An hourglass pointer is preferably used if the delay is less than 10 seconds.

[31]   If there are multiple milestones on a long process, text updates are preferably used to keep the analyst informed.

[32]   A progress bar should contain a method of halting the task at hand. For this reason, all progress dialogs have either a Cancel or Stop button. Cancel is used if halting the process will result in returning the application to the same state as before the action was taken. Use Stop if the process changes the state of the application as it is performed (for example, deleting files, generating a list, etc.), or if data loss may occur. In the former case, stopping will halt any remaining changes, but will keep those performed up to that point. In the latter case, stopping should be followed with a Warning Alert box informing the analyst that there may be a loss of data.

[33]   Progress Dialogs automatically close when the action being performed is completed.

[34]   Help buttons are preferably supplied within the different configuration forms where applicable.

[35]   Tool tips are used to describe, or provide greater detail, on a widget under the system's pointer immediate location. Tool tips can be associated with widgets such as: buttons, sliders, combo boxes, icons, etc. If there is a keyboard shortcut for activation of the particular widget then the keyboard shortcut should be displayed in the tool tip.

[36]   Tool tips should be used carefully, and only if they add information. The tool tip preferably expands upon the widget's purpose. If a text label associated with the widget is specific enough, the use of a tool tip is not recommended. Tool tips presenting long, complex information (paragraphs of text, lists, a matrix of information, etc.) should be avoided.

[37]   Tool tips should appear after a system pointer hovers over a widget for more than approximately 750 milliseconds and its display should not last for more than 4 seconds.

[38]   Slightly longer of contextual help can be given in a message bar at the bottom left of the view panel as an alternative to the use of tool tips. The analyst is typically given the option to turn this feature off in a preference menu.

[39]   All actions that are required within a configuration form can be mapped to command buttons. There are two different types of command buttons, those located in the command button column at the top, right side of the form and those located next to objects within the form. Command buttons are a fast, direct and simple way for the analyst to interact with the view panel.

[40]   Command buttons are buttons within a configuration form that allow the analyst to undertake operations. Buttons that have general influence are preferably placed vertically in the command button column at the top right side of the view panel. Buttons that influence specific widgets within the view panel should be located proximally to that widget.

[41]   An OK button is used when a dialog box requires the user to input information, make changes, etc., but the information is not submitted on the fly, i.e., the changes do not take affect until the analyst decides to save or employ them by pressing OK. If OK is used, no mnemonic is applied. Actuation of the OK button will result in the changed or entered data within the form being saved and the form being closed.

[42]   A dialog window can be used to ask the analyst if the "OK" action should be undertaken as well as to indicate if changes were made to the object being configured simultaneously by another analyst.

[43]   In configuration forms the Save button is used to submit changed or new data entered into a form without closing the form.

[44]   The Close button is used in a dialog box that does not require the analyst to input any information, or the input is performed on the fly. Examples would be in an information alert box that just displays a text message to the user, or a dialog that contains a slider bar whose values are translated to the main application on the fly.

[45]   Cancel is used in the same dialog boxes where an OK button occurs. It closes the window without applying the user's changes or input. Cancel does not have a mnemonic applied to it.

[46]   The Help command button launches context sensitive online documentation.

[47]   A command button can be set as the default button for a configuration form. That is, pressing return will trigger the button. Default buttons add speed and convenience to dialog boxes. However, the use of a default button is not recommended if the action is a destructive one. Default buttons do not have mnemonics. Default buttons have a thicker border surrounding them.

[48]   Command buttons which when actuated open view panels, should have ellipses, placed after their label text.

[49]   Command buttons with ellipses are appropriate for opening view panels that the analyst can choose something from, such as selecting path endpoints. Also, they can be used for bringing up centralized functions, such as connection scheduling or filtering resources for network information. However, command buttons with ellipses are not intended to be replacements for tabs within configuration forms.

[50]   Stand alone command buttons are located next to the objects in the view panel which they directly influence. If a command button only influences one item within the view panel then it should be located next to that item.

[51]   Listed below are a selection of widgets.

[52] An editable field is used to input alphanumeric data into a configuration form. Once the text field is clicked or tabbed to, a blinking insertion point indicator appears within the field.

[53] The editable field is also used for strings of editable text.

[54] A non-editable text field is used to show alphabetical, numeric, and alpha-numeric data to the analyst. The analyst can copy the information within the field, but the field cannot be modified.

[55] Non-editable text fields are used for strings of read only textual information that are generally less than 40 characters long.

[56] A text area is used to input large amounts of alphanumeric data into a configuration form. A text box size used should be wider than it is tall (a rectangle) and of a useful width (30 characters) so that it can be read without spreading the information over too many lines. The text area is typically used when many sentences or more than one line of text are required.

[57] A slider scale is a visual way for the analyst to select a value within a configuration form which usually is qualitative in nature. The slider should be made up of a label, followed by a slider, and finally followed by a editable field (which is optional depending on the nature of the values being entered). The slider scale is typically used to select a value from a range of values or states.

[58] A table is a titled grid of information used to show large sets of data which are required to be entered into a configuration form.

[59] A check box list is composed of an over all label positioned next to a series of check boxes that are each associated with a label.

[60] The check box is a preferred way of selecting multiple items from a list. It provides the analyst with a flat view of all the options facilitating utility and expediency within the view panel.

[61] A radio button is the preferred way of selecting a single item from a list. It provides the analyst with a flat view of all the options facilitating utility and expedience within the view panel.

[62] A scroll pane enables the analyst to control the view of what is being displayed at any time. It consists of a vertical or horizontal "channel" through which a scroll box is moved. Each end of the channel has a scroll arrow.

[63] Configuration form scroll bars are used only within widgets (e.g. combo boxes etc.)

[64] A group box is a non-interactive widget which associates related items within configuration forms (e.g. they all control the same data network entity). The group box has a label and a border.

[65] Group boxes should only be used where a distinct group of items require separation from the rest of the form. Group boxes should almost never be used to surround one item.

[66] A list box is a box containing items which can be selected or multi-selected. It can contain check boxes or radio buttons as well as scroll bars. The list box is used when the analyst is required to make multiple selections of a list of options.

[67] A combo box is an editable or non-editable value field that has a drop-down arrow and drop-down list of possible selections associated with it.

[68] Following are typical uses of the combo box:

– The Combo Box is used to select a single item from a group of items.

– In non-editable combo box's choice is limited to the elements in the drop-down list.

10

– In editable combo boxes the analyst can choose from the drop-down list or enter a unique value in the field. Providing a list of typical selections speeds up the time it takes for the analyst to complete configuration forms.

[69] A spin box is a widget which allows the user to select a value for a field by clicking on up and/or down arrows which are associated therewith.

[70] Spin boxes may have the following drawbacks if used incorrectly:

– Spin boxes increase the chance of errors.

– The up and down arrows are small, hard to hit targets.

– The analyst is prone to getting lost if the set of data that is being selected through the spin box is too large.

[71] A Tab is a way in which the form developer can add multiple panes of information in one configuration form. Tabs preferably should run along the top of the areas that they affect. The tab that is being viewed is to be shown connected to the pane, while the ones not being viewed are in the background. The analyst is required to click on a tab to view the contents of the associated view pane. All changes made on multiple tabs will be saved upon the analyst's confirmation by clicking the "OK" (or "Apply", "Save" etc.) command button.

[72] When a series of distinct groups of objects to configure exist and there is not enough room to put them all on one pane, tabs can be used to break up the groups and allow the analyst to inspect and configure them one at a time.

[73] A Paste button is used to paste the Selected Object from the Selected Object Window into an area. The Paste button is used in any situation that requires the analyst to paste information from one area to another.

United States Patent & Trademark Office
Office of Initial Patent Examination -- Scanning Division
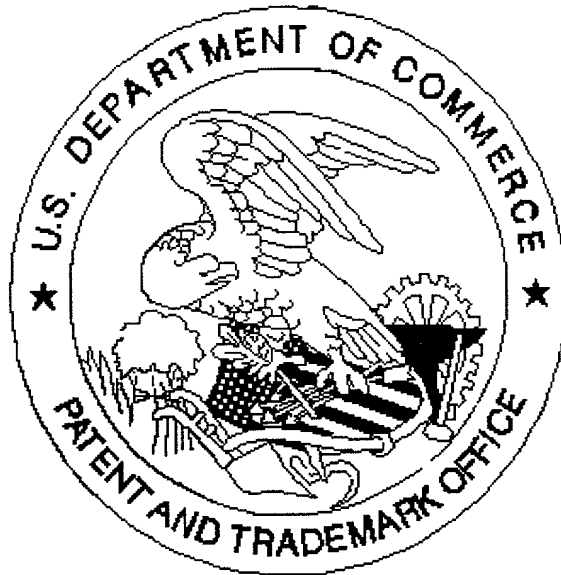
Application deficiencies found during scanning:

☐ Page(s)_____ of _____ were not present
for scanning.   (Document title)

☐ Page(s)_____ of _____ were not
    present
for scanning.   (Document title)

There are 26 pages of specification, not 37 sheets.

☐ *Scanned copy is best available.*